

CLAIMS:

- 1 1. A method comprising:
2 executing a first instruction in a programmable processor to set a rounding mode;
3 and
4 executing a second instruction within the programmable processor to generate an
5 integer result rounded according to the rounding mode.
- 1 2. The method of claim 1, wherein executing the second instruction comprises
2 executing an instruction that performs a rounded averaging operation.
- 1 3. The method of claim 1, wherein executing the second instruction comprises
2 executing an instruction that performs a non-saturating, fixed-point fractional
3 multiplication operation with rounding.
- 1 4. The method of claim 1, wherein executing the second instruction comprises
2 executing an instruction that performs a right-shift operation with rounding.
- 1 5. The method of claim 1, wherein executing the first instruction comprises
2 executing an instruction that sets a rounding mode selected from a group of rounding
3 modes comprising:
4 rounding toward negative infinity;
5 rounding toward infinity;
6 rounding toward zero;
7 rounding away from zero;
8 rounding to a nearest integer, with a value of one-half being rounded
9 towardtoward negative infinity;
10 rounding to a nearest integer, with a value of one-half being rounded
11 towardtoward infinity;
12 rounding to a nearest integer, with a value of one-half being rounded
13 towardtoward zero; and

14 rounding to a nearest integer, with a value of one-half being rounded away from
15 zero.

1 6. A method comprising:
2 performing an operation within a programmable processor to produce a result;
3 adding a rounding term to the result to obtain an intermediate result, the rounding
4 term determined at least in part as a function of a rounding mode, a shift amount, and a
5 sign of the result of the operation; and
6 right-shifting the intermediate result by the shift amount.

1 7. The method of claim 6, further comprising executing an instruction that sets the
2 rounding mode.

1 8. The method of claim 7, wherein executing the instruction comprises executing an
2 instruction that sets a rounding mode selected from the group of rounding modes
3 comprising:
4 rounding towardtoward negative infinity;
5 rounding toward infinity;
6 rounding toward zero;
7 rounding away from zero;
8 rounding to a nearest integer, with a value of one-half being rounded toward
9 negative infinity;
10 rounding to a nearest integer, with a value of one-half being rounded toward
11 infinity;
12 rounding to a nearest integer, with a value of one-half being rounded toward zero;
13 and
14 rounding to a nearest integer, with a value of one-half being rounded away from
15 zero.

1 9. The method of claim 6, wherein the operation is a rounded averaging operation of
2 two or four unsigned byte vectors.

1 10. The method of claim 6, wherein the operation is a non-saturating fixed-point
2 fractional multiplication operation with rounding of a set of vector operands selected
3 from signed half-word vectors, unsigned half-word vectors, signed word vectors and
4 unsigned word vectors.

1 11. The method of claim 6, wherein the operation is an arithmetic right-shift
2 operation on a vector operand by an immediate shift amount with rounding, and wherein
3 the vector operand comprises a signed byte vector, an unsigned byte vector, a signed
4 double word or an unsigned double word.

1 12. A method of compiling a software program comprising parsing the software
2 program to produce instructions executable by a programmable processor, wherein the
3 instructions include a first instruction that sets a rounding mode, and a second instruction
4 that performs an arithmetic operation yielding an integer result rounded according to the
5 rounding mode.

1 13. The method of claim 12, wherein the second instruction performs a rounded
2 averaging operation.

1 14. The method of claim 12, wherein the second instruction performs a non-saturating
2 fixed-point fractional multiplication operation with rounding.

1 15. The method of claim 12, wherein the second instruction performs a right-shift
2 operation with rounding.

1 16. The method of claim 12, wherein the first instruction sets a rounding mode
2 selected from the group of rounding modes comprising:

3 rounding toward negative infinity;

4 rounding toward infinity;

5 rounding toward zero;

6 rounding away from zero;

7 rounding to a nearest integer, with a value of one-half being rounded toward

8 negative infinity;

9 rounding to a nearest integer, with a value of one-half being rounded toward
10 infinity;
11 rounding to a nearest integer, with a value of one-half being rounded toward zero;
12 and
13 rounding to a nearest integer, with a value of one-half being rounded away from
14 zero.

1 17. A method of compiling a software program comprising parsing the software
2 program to produce instructions executable by a programmable processor, wherein the
3 instructions cause the programmable processor to:

4 perform an arithmetic operation;
5 add a rounding term to a result of the arithmetic operation to obtain an
6 intermediate result, the rounding term determined at least in part as a function of a
7 rounding mode, a shift amount, and a sign of the result of the arithmetic operation; and
8 right-shift the intermediate result by the shift amount.

1 18. The method of claim 17, wherein the software program further comprises
2 processor-executable instructions for executing an instruction that sets the rounding
3 mode.

1 19. The method of claim 18, wherein executing the instruction comprises executing
2 an instruction that sets a rounding mode selected from the group of rounding modes
3 comprising:

4 rounding toward negative infinity;
5 rounding toward infinity;
6 rounding toward zero;
7 rounding away from zero;
8 rounding to a nearest integer, with a value of one-half being rounded toward
9 negative infinity;
10 rounding to a nearest integer, with a value of one-half being rounded toward
11 infinity;

12 rounding to a nearest integer, with a value of one-half being rounded toward zero;
13 and
14 rounding to a nearest integer, with a value of one-half being rounded away from
15 zero.

1 20. The method of claim 17, wherein the arithmetic operation is a rounded averaging
2 operation of two or four unsigned byte vectors.

1 21. The method of claim 17, wherein the arithmetic operation is a non-saturating
2 fixed-point fractional multiplication operation with rounding of a set of vector operands
3 selected from signed half-word vectors, unsigned half-word vectors, signed word vectors
4 and unsigned word vectors.

1 22. The method of claim 17, wherein the arithmetic operation is an arithmetic right-
2 shift operation on a vector operand by an immediate shift amount with rounding, wherein
3 the vector operand comprises a signed byte vector, an unsigned byte vector, a signed
4 double word or an unsigned double word.

1 23. A processor-readable medium having processor-executable instructions for:
2 executing a first instruction in a programmable processor to set a rounding mode;
3 and
4 executing a second instruction within the programmable processor to generate an
5 integer result rounded according to the rounding mode.

1 24. The processor-readable medium of claim 23, wherein executing the second
2 instruction comprises executing an instruction that performs a rounded averaging
3 operation.

1 25. The processor-readable medium of claim 23, wherein executing the second
2 instruction comprises executing an instruction that performs a non-saturating fixed-point
3 fractional multiplication operation with rounding.

1 26. The processor-readable medium of claim 23, wherein executing the second
2 instruction comprises executing an instruction that performs a right-shift operation with
3 rounding.

1 27. The processor-readable medium of claim 23, wherein executing the first
2 instruction comprises executing an instruction that sets a rounding mode selected from
3 the group of rounding modes comprising:

4 rounding toward negative infinity;

5 rounding toward infinity;

6 rounding toward zero;

7 rounding away from zero;

8 rounding to a nearest integer, with a value of one-half being rounded toward
9 negative infinity;

10 rounding to a nearest integer, with a value of one-half being rounded toward
11 infinity;

12 rounding to a nearest integer, with a value of one-half being rounded toward zero;

13 and

14 rounding to a nearest integer, with a value of one-half being rounded away from
15 zero.

1 28. A processor-readable medium having processor-executable instructions for:
2 performing an arithmetic operation;
3 adding a rounding term to a result of the arithmetic operation to obtain an
4 intermediate result, the rounding term determined at least in part as a function of a
5 rounding mode, a shift amount, and a sign of the result of the arithmetic operation; and
6 right-shifting the intermediate result by the shift amount.

1 29. The processor-readable medium of claim 28, further comprising processor-
2 executable instructions for executing an instruction that sets the rounding mode.

1 30. The processor-readable medium of claim 29, wherein executing the instruction
2 comprises executing an instruction that sets a rounding mode selected from the group of
3 rounding modes comprising:

4 rounding toward negative infinity;

5 rounding toward infinity;

6 rounding toward zero;

7 rounding away from zero;

8 rounding to a nearest integer, with a value of one-half being rounded toward
9 negative infinity;

10 rounding to a nearest integer, with a value of one-half being rounded toward
11 infinity;

12 rounding to a nearest integer, with a value of one-half being rounded toward zero;

13 and

14 rounding to a nearest integer, with a value of one-half being rounded away from
15 zero.

1 31. The processor-readable medium of claim 28, wherein the arithmetic operation is a
2 rounded averaging operation of two or four unsigned byte vectors.

1 32. The processor-readable medium of claim 28, wherein the arithmetic operation is a
2 non-saturating fixed-point fractional multiplication operation with rounding of a set of
3 vector operands selected from signed half-word vectors, unsigned half-word vectors,
4 signed word vectors and unsigned word vectors.

1 33. The processor-readable medium of claim 28, wherein the arithmetic operation is
2 an arithmetic right-shift operation on a vector operand by an immediate shift amount with
3 rounding, wherein the vector operand comprises a signed byte vector, an unsigned byte
4 vector, a signed double word or an unsigned double word.

1 34. A processor, comprising:

2 a control register to store a rounding mode of a first instruction;

3 a functional unit; and

4 a control unit to direct the functional unit to perform an arithmetic function
5 according to the rounding mode in response to a second instruction.

1 35. The processor of claim 34, wherein the second instruction comprises a rounded
2 averaging operation.

1 36. The processor of claim 34, wherein the second instruction comprises a non-
2 saturating fixed-point fractional multiplication operation with rounding.

1 37. The processor of claim 34, wherein the second instruction comprises a right-shift
2 operation with rounding.

1 38. The processor of claim 34, further comprising:
2 a fetch unit configured to receive an instruction from an instruction stream;
3 a decode unit configured to decode the received instruction; and
4 a register file coupled to the plurality of functional units and configured to store
5 the integer result.

1 39. The processor of claim 34, wherein the first instruction sets a rounding mode
2 selected from the group of rounding modes comprising:
3 rounding toward negative infinity;
4 rounding toward infinity;
5 rounding toward zero;
6 rounding away from zero;
7 rounding to a nearest integer, with a value of one-half being rounded toward
8 negative infinity;
9 rounding to a nearest integer, with a value of one-half being rounded toward
10 infinity;
11 rounding to a nearest integer, with a value of one-half being rounded toward zero;
12 and
13 rounding to a nearest integer, with a value of one-half being rounded away from
14 zero.

1 40. A processor, comprising:

2 a control unit comprising a control register configured to store a representation of
 3 a selected rounding mode;
 4 at least one functional unit coupled to the control register;
 5 a fetch unit configured to receive an instruction from an instruction stream;
 6 a decode unit configured to decode the received instruction; and
 7 a register file coupled to the plurality of functional units,
 8 the control unit configured to
 9 perform an arithmetic operation,
 10 add a rounding term to a result of the arithmetic operation to obtain an
 11 intermediate result, the rounding term determined at least in part as a function of the
 12 selected rounding mode, a shift amount, and a sign of the result of the arithmetic
 13 operation,
 14 right-shift the intermediate result by the shift amount to generate a
 15 rounded result, and
 16 store the rounded result in the register file.

1 41. The processor of claim 40, wherein the control unit is further configured to
 2 execute an instruction that sets the rounding mode.

1 42. The processor of claim 41, wherein the instruction sets a rounding mode selected
 2 from the group of rounding modes comprising:
 3 rounding toward negative infinity;
 4 rounding toward infinity;
 5 rounding toward zero;
 6 rounding away from zero;
 7 rounding to a nearest integer, with a value of one-half being rounded toward
 8 negative infinity;
 9 rounding to a nearest integer, with a value of one-half being rounded toward
 10 infinity;
 11 rounding to a nearest integer, with a value of one-half being rounded toward zero;
 12 and

13 rounding to a nearest integer, with a value of one-half being rounded away from
14 zero.

1 43. The processor of claim 40, wherein the arithmetic operation is a rounded
2 averaging operation of two or four unsigned byte vectors.

1 44. The processor of claim 40, wherein the arithmetic operation comprises
2 performing a non-saturating fixed-point fractional multiplication operation with rounding
3 of a set of vector operands selected from signed half-word vectors, unsigned half-word
4 vectors, signed word vectors and unsigned word vectors.

1 45. The processor of claim 40, wherein the arithmetic operation is an arithmetic right-
2 shift operation on a vector operand by an immediate shift amount with rounding, wherein
3 the vector operand comprises a signed byte vector, an unsigned byte vector, a signed
4 double word or an unsigned double word.

1 46. A system comprising:
2 a memory; and
3 a processor comprising
4 a control register to store a rounding mode of a first instruction,
5 a functional unit, and
6 a control unit to direct the functional unit to perform an arithmetic
7 function according to the rounding mode in response to a second instruction.

1 47. The system of claim 46, wherein the second instruction comprises a rounded
2 averaging operation.

1 48. The system of claim 46, wherein the second instruction comprises a non-
2 saturating fixed-point fractional multiplication operation with rounding of a set of vector
3 operands selected from signed half-word vectors, unsigned half-word vectors, signed
4 word vectors and unsigned word vectors.

1 49. The system of claim 46, wherein the second instruction comprises a right-shift
2 operation with rounding.

1 50. The system of claim 46, wherein the rounding mode is selected from the group of
2 rounding modes comprising:

3 rounding toward negative infinity;

4 rounding toward infinity;

5 rounding toward zero;

6 rounding away from zero;

7 rounding to a nearest integer, with a value of one-half being rounded toward
8 negative infinity;

9 rounding to a nearest integer, with a value of one-half being rounded toward
10 infinity;

11 rounding to a nearest integer, with a value of one-half being rounded toward zero;

12 and

13 rounding to a nearest integer, with a value of one-half being rounded away from
14 zero.

1 51. A system, comprising:

2 a memory; and

3 a processor coupled to access the memory, the processor comprising

4 a control unit comprising a control register configured to store a
5 representation of a selected rounding mode,

6 at least one functional unit coupled to the control register,

7 a fetch unit configured to receive an instruction from an instruction
8 stream,

9 a decode unit configured to decode the received instruction, and

10 a register file coupled to the plurality of functional units,

11 the control unit configured to

12 perform an arithmetic operation,

13 add a rounding term to a result of the arithmetic operation to obtain
14 an intermediate result, the rounding term determined at least in part as a function of the

15 selected rounding mode, a shift amount, and a sign of the result of the arithmetic

16 operation,

17 right-shift the intermediate result by the shift amount to generate a
18 rounded result, and
19 store the rounded result in the register file.

1 52. The system of claim 51, wherein the processor is further configured to execute an
2 instruction that sets the rounding mode.

1 53. The system of claim 52, wherein executing the instruction comprises executing an
2 instruction that sets a rounding mode selected from the group of rounding modes
3 comprising:

4 rounding toward negative infinity;
5 rounding toward infinity;
6 rounding toward zero;
7 rounding away from zero;
8 rounding to a nearest integer, with a value of one-half being rounded toward
9 negative infinity;
10 rounding to a nearest integer, with a value of one-half being rounded toward
11 infinity;
12 rounding to a nearest integer, with a value of one-half being rounded toward zero;
13 and
14 rounding to a nearest integer, with a value of one-half being rounded away from
15 zero.

1 54. The system of claim 51, wherein the arithmetic operation is a rounded averaging
2 operation of two or four unsigned byte vectors.

1 55. The system of claim 51, wherein the arithmetic operation is a non-saturating
2 fixed-point fractional multiplication operation with rounding of a set of vector operands
3 selected from signed half-word vectors, unsigned half-word vectors, signed word vectors
4 and unsigned word vectors.

1 56. The system of claim 51, wherein the arithmetic operation is an arithmetic right-
2 shift operation on a vector operand by an immediate shift amount with rounding, wherein

- Case 22300-05725